

مقدمه‌ای بر بهینه‌سازی رویه‌های بازیابی در سیستم مدیریت پایگاه داده‌های رابطه‌ای

ارائه شده به:

استاد روحانی رانکوهی

توسط:

محمد شفیع

پایگاه داده

ترم اول، سال ۸۵-۸۴

دانشکده مهندسی کامپیوتر

دانشگاه صنعتی شریف

پیشگفتار

در این مقال تنها بر آنیم که به گزارش بعضی مفاهیم و راه‌های ممکن و معمول در بهینه‌سازی رویه‌های بازیابی^۱ که توسط دیگران ارائه شده است بپردازیم و در مواردی تنها به اشاره موردی به مطلب مورد نظر و بیانی اجمالی و ذکر منابع مورد مطالعه بسنده می‌کنیم، که با توجه به گستردگی مبحث و وسعت مسائل مطرح شده در این زمینه، مطالعه و تحقیقی در خور ادای مطلب نیاز به زمان بیشتر دارد و به وضوح در حوصله این چند صفحه نمی‌گنجد.

با توجه به اهمیت زمان و سرعت دسترسی به اطلاعات در عصر حاضر، توجه به این مطلب و مطالعه در زمینه‌ی آن به خصوص از دید عملی می‌تواند بسیار با اهمیت باشد و با در نظر گرفتن مدل رابطه‌ای و انتزاع آن در سطح تئوری و استفاده‌هایی که از این جنبه در بهینه‌سازی قابل طرح است، این مبحث از جذابیت نسبی در بعد تئوریک هم برخوردار است و در هر دو بعد دارای شاخه‌ها و زمینه‌های فعالیت گوناگون می‌باشد.

ما در این میان همانطور که پیشتر بیان شد بیشتر به بیان اجمالی مطالب می‌پردازیم و در انتها مروری بیشتر در مورد بهینه‌سازی رویه‌ها از لحاظ معنایی خواهیم داشت. به این امید که در حد خود مورد قبول خواننده اهل واقع شود.

^۱ query

فهرست مطالب

۱.....	مقدمه‌ای بر بهینه‌سازی رویه‌های بازیابی در سیستم مدیریت پایگاه داده‌های رابطه‌ای
۲.....	پیشگفتار
۴.....	چکیده
۴.....	مقدمه
۴.....	مراحل پردازش رویه‌ها
۶.....	راه‌های مختلف بهینه‌سازی رویه‌ها
۶.....	بهینه‌سازی رویه‌ها از لحاظ معنایی
۷.....	چند نکته در استفاده از بهینه‌سازی معنایی
۷.....	نتیجه‌گیری
۷.....	خلاصه
۸.....	پیوستها
۹.....	پیوست ۱
۱۰.....	فهرست منابع

چکیده

در این تحقیق در ابتدا به توضیح مختصری در مورد بهینه‌سازی و مراحل مختلف در پردازش رویه‌های بازیابی در مسیر بهینه‌سازی می‌پردازیم و سپس به اجمال راه‌های مختلف بهینه‌سازی رویه‌ها را بررسی می‌کنیم و در نهایت شرح مختصری در مورد بهینه‌سازی رویه‌ها از طریق معنایی خواهیم داشت.

مقدمه

در کنار سایر مفاهیم مربوط به پایگاه داده‌ها، بهینه‌سازی در انجام رویه‌های بازیابی اطلاعات از اهمیت در خور توجهی در زمینه‌های عملی برخوردار است؛ چرا که تمام فعالیتها در نهایت برای استفاده از این تکنولوژی در عمل می‌باشد و در عمل هم دسترسی هر چه سریعتر به اطلاعات اهمیت ویژه‌ای دارد. البته اعمال سیاست‌های بهینه‌سازی نباید به نحوی باشد که برای کسانی که نمی‌خواهند از این امکان استفاده کنند هزینه افزوده‌ای در بر داشته باشد و شاید بتوان گفت که توجه به این امر یک اصل است.

در مورد سیستم‌های رابطه‌ای بهینه‌سازی هم بیانگر یک رقابت^۱ است و هم یک فرصت^۲. یک رقابت و یا به بیانی یک مبارزه‌طلبی است چون در هر سیستمی که می‌خواهد کارایی قابل قبولی داشته باشد لازم می‌نماید و بهینه‌سازی و اعلام سیاست‌های مربوط به آن هم سختی و مشکلات خاص خود را دارد؛ و اما یک فرصت است زیرا یکی از نقاط قوت در مشی و مدل رابطه‌ای برای پایگاه‌های داده‌ها انتزاع بالای معنایی است که تأمین می‌کند و در قدم اول، بهینه‌سازی رویه‌های پاسخگو را تسهیل می‌کند. [DATE]

آنچه بیانگر منظور ما از بهینه‌سازی در این میان می‌باشد، همان بهینه‌سازی خودکار است که توسط خود DBMS و بدون دخالت کاربر انجام می‌شود به نوبه خود مزایایی دارد. [DATE]

مراحل پردازش رویه‌ها

مراحل پردازش و بررسی رویه‌ها در یک سیستم مدیریت پایگاه داده را بعد از بررسی از لحاظ دستوری و اطمینان از صحت آن می‌توان به صورت زیر بیان کرد: [DATE]

تبدیل رویه به فرم درونی و نمایش آن با نمودار درختی^۳: این مرحله تبدیلات بین سطوح مختلف در معماری پایگاه داده‌ها تا سطح درونی انجام و در نهایت رویه به صورت یک نمودار درختی و یا گراف^۴ [ELMA] نمایش داده می‌شود.

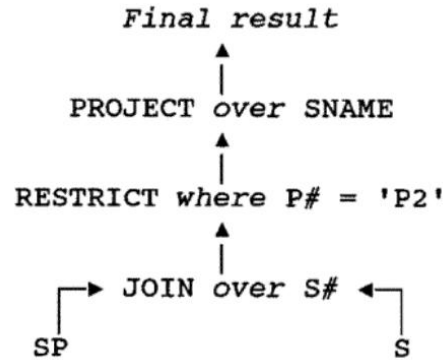
^۱ challenge

^۲ opportunity

^۳ query tree

^۴ query graph

در شکلی که در پی می‌آید نمونه‌ای از یک رویه و نمودار درختی آن آمده است. پایگاه داده مورد استفاده همان پایگاه داده معروف "قطعه - تهیه کننده" است.



شکل ۱ نمودار درختی ((SP JOIN S) WHERE P# = P#('P2')) {SNAME}

۱. **تبدیل به فرم کانونی^۱**: در این مرحله مجموعه‌ای از تبدیلات روی رویه انجام می‌شود که مستقل از پیاده‌سازی فیزیکی و کاردینالیته روابط موجب بهبود رویه و نوعی بهینه‌سازی می‌شود. مواری مانند ترتیب اعمال عملگرها روی روابط در این مقوله جای می‌گیرند. مثلاً در یک رویه اگر ابتدا عملگر گزینش و سپس عملگر پیوند را اعمال کنیم کاردینالیته رابطه در موقع پیوند کمتر و به طبع رویه بهینه‌تر خواهد بود تا این که به صورت عکس عمل شود. (برای اطلاع بیشتر از خصوصیات عملگرهای جبر رابطه‌ای به [روحا] مراجعه کنید.) در واقع یکی از خصوصیات مدل رابطه‌ای این است که هر عمل یکسان را توسط رویه‌های متفاوتی می‌توان انجام داد و در بهینه‌سازی تنها کفایت که رویه‌ها را به یک زیرمجموعه‌ای که بهینه‌تر است و همان عمل را انجام می‌دهد تبدیل و بهینه‌سازی‌های بعدی را فقط روی آن مجموعه انجام دهیم. به آن مجموعه کوچکتر که با تعریف یک ضابطه یکسانی هر رویه در مجموعه رویه‌های یکسان تنها به یک رویه در آن مجموعه نظیر می‌شود، مجموعه فرم‌های کانونی می‌گوییم.

۲. **انتخاب رویه‌های سطح پایین کاندید^۲**: در این مرحله رویه‌های سطح پایین و خصوصیات فیزیکی و روابط مورد توجه قرار می‌گیرند که هر کدام یک هزینه‌ای خواهند داشت که در انتخاب آنها در مرحله بعدی تأثیرگذار است.

۳. **تولید طرح‌های اجرای رویه^۱ و انتخاب کم هزینه‌ترین**: در این مرحله طرح‌های مختلف بررسی و کم هزینه‌ترین طرح انتخاب می‌شود. برای توضیح بیشتر به پیوست ۱ مراجعه کنید.

^۱ canonical form

^۲ candidate low-level procedures

راه‌های مختلف بهینه‌سازی رویه‌ها

در این بخش به بررسی راه‌های مختلف در بهبود اجرای رویه‌ها می‌پردازیم.

۱. **سیاست تقسیم و حل:** در مورد بهینه‌سازی رویه‌های پاسخگو در پایگاه داده‌ها هم همانند سایر مسائل می‌توان از این استراتژی استفاده کرد، به خصوص در مورد رویه‌های تودرتو و شروط پیچیده استفاده از این سیاست کاربرد دارد و موجب تسهیل کار بهینه‌سازی می‌شود.

البته این استراتژی تنها یک شیوه جهت تسهیل کار می‌باشد و طی آن از موارد دیگر استفاده می‌شود و در یک تقسیم‌بندی دیگر شاید اصلاً جزء راه‌های بهینه‌سازی قرار نگیرد.

۲. **استفاده از خصوصیات عملگرها و انتخاب بهترین ترتیب:** شامل مواردی مانند استفاده از خصوصیات عملگرهای جبر و حساب رابطه‌ای و زیرمجموعه بودن روابط و از این دست می‌باشد. (ر.ک. [روحا])

۳. **استفاده از اطلاعات آماری:** توجه به مواردی مانند کاردینالیته و درجه روابط و رابطه نتیجه حاصل و انتخاب ترتیب مناسب مثلاً در پیوند روابط در این دسته جای می‌گیرند. (برای توضیح بیشتر در مورد آخر به پیوست ۱ مراجعه کنید).

۴. **بهره‌گیری از محدودیت‌های جامعیتی پایگاه داده‌ها:** این محدودیت‌ها شامل محدودیت‌های نوع و پایگاهی می‌باشد و استفاده از آنها در بازنویسی رویه‌های بهینه‌تر می‌باشد و به عبارتی بازنویسی رویه با توجه به معنای آن و استفاده از این محدودیت‌های معنایی می‌باشد که در بخش بعدی مورد بررسی قرار می‌دهیم.

بهینه‌سازی رویه‌ها از لحاظ معنایی

یکی از راه‌های کارآمد در بهینه‌سازی رویه‌های بازیابی بررسی رویه از نظر معنایی است مثلاً با یک بررسی معنایی روی رویه زیر واضح است که هیچ جوابی ندارد و دیگر لزومی ندارد آن را اجرا کرد.

```
SELECT * FROM table_name where 1 = 2;
```

البته ملاحظاتی وجود دارد که در ادامه خواهیم دید.

همچنین اگر مقدار مورد درخواست در محدوده دامنه نباشد باز هم با یک بررسی معنایی ولی اینبار با توجه به قواعد جامعیتی پایگاه داده‌ها می‌توان یک بهینه‌سازی در صورت لزوم انجام داد.

موارد پیچیده‌تری هم وجود دارد که در آنها با استفاده از محدودیت‌های صفتی، رابطه‌ای و یا پایگاهی می‌توان یک قاعده معنای استخراج و در بهینه‌سازی از آن استفاده کرد.

در کنار این موارد و استفاده از محدودیت‌های موجود کارهایی نیز در زمینه استفاده از داده‌ها و یا میزان رویه‌ها و اینکه جواب آنها بیشتر در چه محدوده‌ای بوده و یا بر اساس چه توزیع آماری روی داده‌ها بوده است و ارائه الگوریتم‌های یادگیز در این زمینه انجام شده [HSU] که بررسی آنها از حوصله بحث خارج است.

چند نکته در استفاده از بهینه‌سازی معنایی

در زمینه استفاده از بهینه‌سازی معنایی و مطالعه و تحقیق در این زمینه توجه به چند مسئله مهم به نظر می‌رسد.

- در این مورد نقش دقیق محدودیت‌های جامعیتی و میزان استفاده از آنها به طور دقیق مشخص نشده است.
- بیشتر کاری که در این زمینه انجام شده مربوط به DD¹ ها می‌باشد و در مورد سیستم‌های مدیریت پایگاه داده‌های رابطه کار در خور توجهی حداقل در زمینه پیاده‌سازی انجام نشده است.
- چه موقع استفاده از این مطلب به صرفه است و ارزش هزینه آن را دارد؟

(این موارد در [GENE] مطرح و در پیرامون آنها بحث شده است.)

و مواردی مانند مثال ذکر شده در قسمت قبل که آیا رویه اجرا شود و یا خیر و یا به طریق دیگری اجرا شود؛ چه بسا که در مثال بخش قبل فرد می‌خواهد از آن رویه در رویه دیگری در جهت ساخت همان ساختار استفاده کند و یا شاید نتیجه یک اشتباه باشد.

قابل ذکر است که در سیستم‌های مدیریت پایگاه داده‌های رابطه‌ای موجود استفاده از این راه بهینه‌سازی بسیار ناچیز است و یا اصلاً استفاده نشده است.

نتیجه‌گیری

با توجه به مطالبی که در پیش آمد و نیز افزایش روز افزون حجم اطلاعات و به طبع استفاده بیشتر از سیستم‌های مدیریت پایگاه داده‌ها به خصوص سیستم‌های مدیریت پایگاه داده‌های رابطه‌ای و اهمیت سرعت انجام کار و دسترسی به اطلاعات اهمیت کار و تحقیق در این زمینه روشن است و همانطور که به نظر می‌آید کار در زمینه بهینه‌سازی معنایی هم خود شاخه‌های گسترده و رویه رشدی دارد که ورود و فعالیت در آن سودمند می‌نماید.

خلاصه

در این گزارش بعد از مطالعه یک مقدمه مختصر در زمینه بهینه‌سازی به مراحل پردازش رویه‌ها پرداختیم و سپس در ادامه به بررسی راه‌های معمول در بهینه‌سازی پرداختیم و در نهایت توجهمان را معطوف به بهینه‌سازی معنایی کردیم و مطالبی پیرامون آن را بررسی کردیم.

¹ Deductive Database

پيوسنتها

پیوست ۱

در این قسمت تعداد حالات مختلفی که می‌توان حاصل پیوند بین n رابطه را محاسبه کرد مورد بررسی قرار می‌دهیم.

$$R_1 \bowtie R_2 \bowtie \dots \bowtie R_n$$

از آنجایی که جواب یک رابطه است و در رابطه ترتیب مهم نیست می‌توان روابط ۱ تا n را به $n!$ حالت در یک ترتیب برای پیوند قرار داد و تعداد حالاتی که می‌توان یک چنین پیوندی را انجام داد معادل است با:

$$A_{n-1} = A_0 A_{n-2} + A_1 A_{n-3} + \dots + A_{n-2} A_0$$

که A_i بیانگر یک پیوند بین " $i + 1$ " رابطه که شامل " i " پیوند است می‌باشد و بیانگر این مطلب است که یک پیوند با " $n - 1$ " پیوند از طریق پیوند یک رابطه که شامل " $n - 1$ " و یک تک رابطه حاصل می‌شود و به همین ترتیب تمام حالات ممکن در رابطه بالا مشهود است.

کاملاً واضح است که این مقدار برابر حالات مختلف پرانتزگذاری ضرب n فاکتور است که برابر با عدد کاتالان می‌باشد. که در مورد این مسئله برابر است با [JARV]:

$$A_{n-1} = C(2(n-1), n-1) / n = (2(n-1))! / n!(n-1)!$$

که در نتیجه تعداد کل حالات مختلف محاسبه این رابطه عبارت است از:

$$T_{n-1} = (2(n-1))! / (n-1)!$$

که در آن n تعداد روابط است.

با توجه به کثرت حالات انتخاب بهترین ترتیب و ترکیب در انجام پیوندها و محاسبه نتیجه تا حد امکان همواره روابطی که در پیوند شرکت می‌کنند از کاردینالیتی کوچکتری برخوردار باشند می‌تواند بسیار در بهینه‌سازی رویه و سرعت بازیابی تأثیر داشته باشد.

فهرست منابع

1. [DATE]: DATE C.J. . An Introduction to Database Systems. 7th ed. . Addison-Wesley. USA. 2000.
 2. [ELMA]: ELMASRI Ramez, NAVATHE Shamkant B. . Fundamentals of Database Systems. Third ed. . Addison-Wesley. USA. 2000.
 3. [HSU]: HSU Chun-Nan, KNOBLOCK Craig A. . Using Inductive Learning to Generate Rules for Semantic Query Optimization. Advances of Knowledge Discovery and Data Mining. AAAI Press. 1995
 4. [GENE]: GENET Bryan, HINZE Annika. Open Issues in Semantic Query Optimization in Relational DBMS. University of Waikato. New Zealand.
 5. [JARV]: JARVIS Frazer. Catalan Numbers.
۶. [روحا]: روحانی رانکوهی، سید محمد تقی، مفاهیم بنیادی پایگاه داده‌ها، چاپ سوم، انتشارات جلوه، تهران، ۱۳۸۳